Introduction to Data Science

Model Evaluation

Arijit Mondal

Dept. of Computer Science & Engineering

Indian Institute of Technology Patna

arijit@iitp.ac.in

Introduction

- Extracting meaningful information from the past data is one of the major challenges now
- This requires to build efficient model which can be queried to get relevant information
- After developing the model, performance evaluation of the same is also very critical
- There are different methods/approaches for evaluation of a model. It also depends on the problem at hand

Mathematical model

- The purpose is to encapsulate information into a tool
  - The tool can be used to forecast, make prediction, etc
- Predictive model tries to forecast future behavior by observing past data/events
  - Laws of physics are used to provide principled notions of causation
- Primary targets are
  - Design of a model
  - Verify the model
  - Evaluation of model

Best model

- All models are wrong and some are useful. — George Box
- There are many ways to fit a given data

Best model

- All models are wrong and some are useful. — George Box
- There are many ways to fit a given data

- Things to consider while selecting a model
  - Occam's Razor
    - The simplest explanation is the best explanation
    - In other words, simplest model is the best model

Best model

- All models are wrong and some are useful. — George Box
- There are many ways to fit a given data

- Things to consider while selecting a model
  - Occam's Razor
    - The simplest explanation is the best explanation
    - In other words, simplest model is the best model
  - Bias-Variance tradeoff
    - Bias — This error caused from the incorrect assumption of the model
    - Variance — This error resulted from sensitivity to fluctuation in the training set

Signal & Noise

- Think probabilistically
  - Example: India has 23% chance to win the test match
  - Example: India will loose the match
  - One can describe using a distribution also
- Change your forecast in response to new information
  - Live models are better than dead one
  - Maintaining live models is not trivial
- Look for consensus
  - Multiple models should be build to predict the same thing
  - Compare with competing third party forecast
- Employ Bayesian reasoning
  - $P(A|B) = \frac{P(B|A)P(A)}{P(B)}$

Types of models

- Linear vs Non-linear
  - Linear combination of features (eg. Linear regression), easy to fit and explain
  - Higher order polynomial, logarithmic, exponential functions are often required
  - It is harder to fit non-linear model (eg. Deep Learning)
- Black-box vs Descriptive
  - Black-box works in unknown manner (eg. Deep Learning)
  - Descriptive methods provide some insights (eg. Linear regression, Decision Trees)
  - Descriptive models are primarily theory driven
  - ML models are less opaque
  - DL models are often very effective
  - DL model can be fooled also

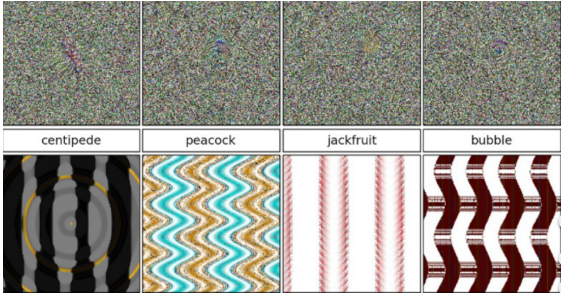


image source: Data Science Design Manual

Types of models (contd)

- First principle vs Data driven
  - First principle relies on law of physics, theoretical rules/laws
  - Data driven models are based on observed correlation between input and outcome variables
- Stochastic vs Deterministic
  - Stochastic is based on randomness
  - It uses probability
  - All rules of probabilities apply
  - Deterministic model yields only one answer and these are based on first principle usually
- Flat vs Hierarchical
  - Many problems exist on several different levels, each of which may require independent submodule (eg. general state of company, balance sheet performance)
  - Hierarchical structure improves a logical and transparent way to build the model
  - Deep learning is a mixed model

Baseline models

- 'A broken clock is right twice a day' !!
- First step is to built a base model - simplest reasonable model that produce answers we can compare with
- More sophisticated models should perform better than base model

Evaluation of models

- Error can results from many things like data normalization, preprocessing, post-processing, etc.
- Check with a few *positive* and *negative* examples
- Typically accuracy is the prime measure
- Performance needs to be measured on unseen data

Evaluation of classifier

- Consider two class classification
- There are four possible scenarios (confusion matrix or contingency table)

Evaluation of classifier

- Consider two class classification
- There are four possible scenarios (confusion matrix or contingency table)
  - TP — classifier labels a positive item as positive, win situation (True Positive)

Evaluation of classifier

- Consider two class classification
- There are four possible scenarios (confusion matrix or contingency table)
  - TP — classifier labels a positive item as positive, win situation (True Positive)
  - TN — classifier correctly labels a negative item as negative, win situation (True Negative)

Evaluation of classifier

- Consider two class classification
- There are four possible scenarios (confusion matrix or contingency table)
  - TP — classifier labels a positive item as positive, win situation (True Positive)
  - TN — classifier correctly labels a negative item as negative, win situation (True Negative)
  - FP — classifier labels a negative item as positive, Type I error, (False Positive)

Evaluation of classifier

- Consider two class classification
- There are four possible scenarios (confusion matrix or contingency table)
  - TP — classifier labels a positive item as positive, win situation (True Positive)
  - TN — classifier correctly labels a negative item as negative, win situation (True Negative)
  - FP — classifier labels a negative item as positive, Type I error, (False Positive)
  - FN — classifier mistakenly declares labels a positive item as negative, Type II error, (False Negative)

Accuracy, Precision

- Accuracy $= \dfrac{TP + TN}{TP + TN + FP + FN}$

Accuracy, Precision

- Accuracy $= \dfrac{TP + TN}{TP + TN + FP + FN}$
  - Accuracy measure has some issue

Accuracy, Precision

- Accuracy $= \dfrac{TP + TN}{TP + TN + FP + FN}$
  - Accuracy measure has some issue
  - Let us assume that 5% of the particular test takers really had the disease (positive class). One can predict all examples to be negative

Accuracy, Precision

- Accuracy $= \dfrac{TP + TN}{TP + TN + FP + FN}$
  - Accuracy measure has some issue
  - Let us assume that 5% of the particular test takers really had the disease (positive class). One can predict all examples to be negative
  - This can lead to accuracy 95%
- To overcome this ie., more sensitive to getting to positive class right we use

  Precision $= \dfrac{TP}{TP + FP}$

Accuracy, Precision

- Accuracy $= \dfrac{TP + TN}{TP + TN + FP + FN}$
  - Accuracy measure has some issue
  - Let us assume that 5% of the particular test takers really had the disease (positive class). One can predict all examples to be negative
  - This can lead to accuracy 95%
- To overcome this ie., more sensitive to getting to positive class right we use

  Precision $= \dfrac{TP}{TP + FP}$
  - If there are less positive samples, so classifier achieves low TP
  - In medical diagnosis case, one may tolerate FP but not FN

Recall, F-score

- We use recall - how often one is right on all positive examples -

$$\text{Recall} = \frac{TP}{TP + FN}$$

Recall, F-score

- We use recall - how often one is right on all positive examples -

  $$\text{Recall} = \frac{TP}{TP + FN}$$

- To have a single measure, we use F-score, it is defined as

  $$\text{F-score} = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

Recall, F-score

- We use recall - how often one is right on all positive examples -

  $$\text{Recall} = \frac{TP}{TP + FN}$$

- To have a single measure, we use F-score, it is defined as

  $$\text{F-score} = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

  - Harmonic mean is less than arithmetic mean
  - Lower number has a disproportionate large effect

Balanced classifier

- A classifier that performs equally good in both positive and negative examples
- Consider a set of n items of which $p \cdot n$ are of positive examples and $(1 - p) \cdot n$ negative
- Consider a random classifier that predicts positive class correctly with probability q
- Also, the expected performance of a balanced classifier, which somehow correctly classifies members of each class with probability q

Balanced classifier

- A classifier that performs equally good in both positive and negative examples
- Consider a set of n items of which $p \cdot n$ are of positive examples and $(1-p) \cdot n$ negative
- Consider a random classifier that predicts positive class correctly with probability q
- Also, the expected performance of a balanced classifier, which somehow correctly classifies members of each class with probability q

	Random Classifier		Balanced Classifier	
	Predicted class		Predicted class	
	yes	no	yes	no
yes	$(pn)q$	$(pn)(1-q)$	$(pn)q$	$(pn)(1-q)$
no	$((1-p)n)q$	$((1-p)n)(1-q)$	$((1-p)n)(1-q)$	$((1-p)n)q$

Example

- Fill the following table for the following scenario (disease detection)
- The people who have undergone a test diagnosed with no-disease 95% cases and disease with 5% scenarios
- A 'sharp' classifier always says a fixed outcome

	Random		Sharp		Balanced		
q	0.05	0.5	0.0	1.0	0.5	0.9	1.0

Example

- Fill the following table for the following scenario (disease detection)
- The people who have undergone a test diagnosed with no-disease 95% cases and disease with 5% scenarios
- A 'sharp' classifier always says a fixed outcome

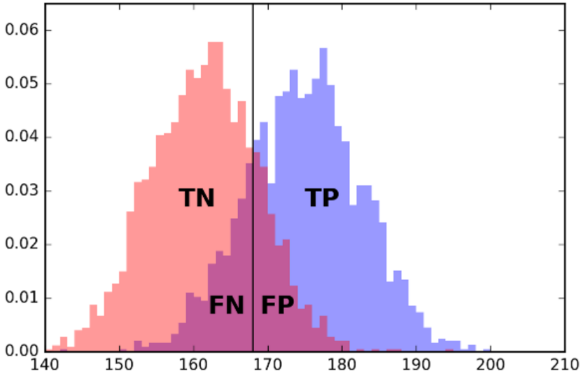
	Random		Sharp		Balanced		
q	0.05	0.5	0.0	1.0	0.5	0.9	1.0
accuracy	0.905	0.5	0.95	0.05	0.5	0.9	1.0
precision	0.05	0.05	—	0.05	0.05	0.321	1.0
recall	0.05	0.5	0	1.0	0.5	0.9	1.0
F-score	0.05	0.091	—	0.095	0.091	0.474	1.0

Observations

- Accuracy is a misleading when the class sizes are substantially different
- Recall equals accuracy if and only if the classifiers are balanced
- High precision is very hard to achieve in unbalanced class sizes
- F-score does the best job of any single statistics but all four work together to describe the performance of a classifier

ROC curve

- Receiver-Operator Characteristic (ROC) curve



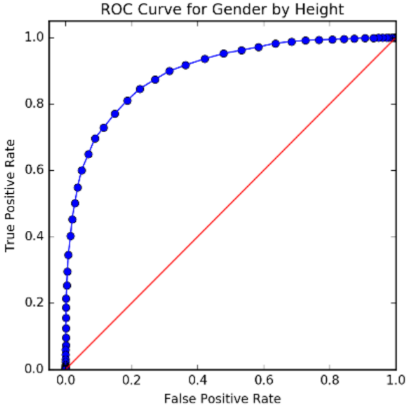


image source: Data Science Design Manual

Evaluating multiclass systems

- Consider a news classification model that categorizes news into d classes
- Expected accuracy for a random classifier is $1/d$
- Accuracy drops rapidly with increased class complexity
- A better measure is the *top-k success rate*
- Precision and recall are defined as follows

$$precision_i = C_{ii} / \sum_{j=1}^{d} C_{ji}$$

$$recall_i = C_{ii} / \sum_{j=1}^{d} C_{ij}$$

where C_{ij} denotes how many items of class i labeled as j

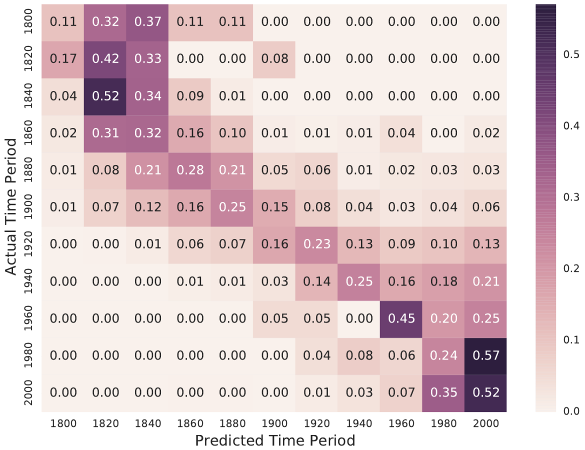
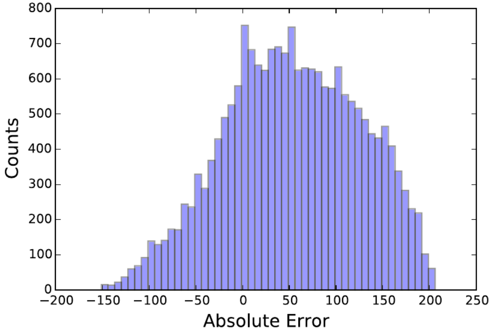


image source: Data Science Design Manual

Evaluating value prediction models

- It can also be thought of classification however there are infinite class
- Error statistics
  - Error is a function of the difference between forecast and actual result
  - Measuring the performance of a value prediction system involves the following
    - Fixing the specific individual error function
    - Selecting that statistics to best represent the full error distribution
- Choices for error function (predicted - y', actual - y)
  - Absolute error: $\|y - y'\|$. It is the difference between actual and predicted values. No sign is considered.
  - Relative error: $\dfrac{y - y'}{y}$
  - Squared error: $(y' - y)^2$
- Histogram of the absolute error distribution may be looked into
- The distribution should be symmetric and centered around 0, also, it should be bell shaped

Error Histogram example



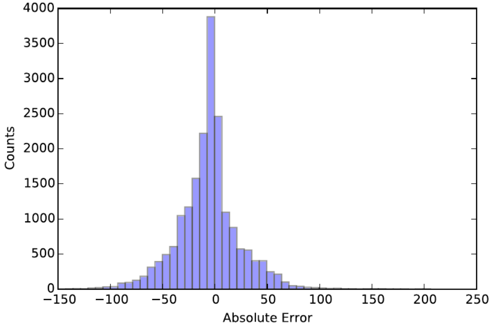


image source: Data Science Design Manual

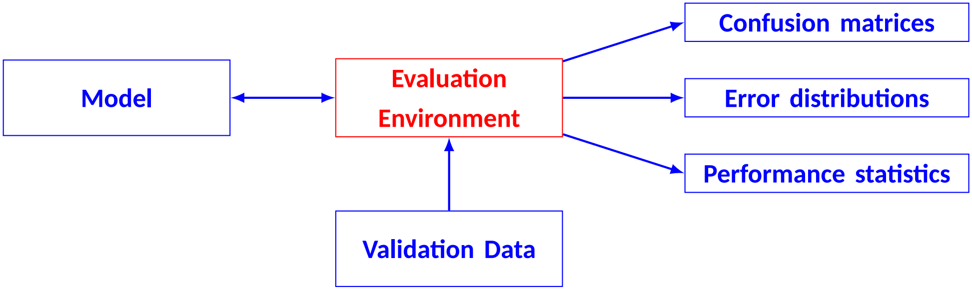
Summary statistics

- Error distribution needs to be reduced to a single number in order to compare the performance of different value prediction models
- Commonly used metric is *mean squared error* (MSE)

$$MSE(Y, Y') = \frac{1}{n}\sum_{i=1}^{n}(y_i' - y_i)^2$$

- Other choice is root mean squared - $RMSD = \sqrt{MSE(Y, Y')}$

Model evaluation environment



Data hygiene for evaluation

- Training data — Used for building the model
- Validation data — Used for learning hyper-parameters
- Test data — Used for testing of the model

Amplifying small data sets

- Cross validation — Typically used when the dataset is limited
  - Partition the data into k equal-sized chunks, then trains k models
  - Model i is trained on the union of all blocks $x \neq i$, totaling $(k-1)/k$th of the data
  - Model is tested on the held out ith block
  - Average performance of these k classifiers is considered as full model
- Perturb real examples to create similar but synthetic ones
  - Add noise, Data augmentation
- Give partial credit
  - Transcription

Summary

- Good performance on data you trained models on is very suspect, because models can easily be overfit
- Model should perform well on unseen data
- Appropriate metric needs to be chosen